UNITED STATES DISTRICT COURT
SOUTHERN DISTRICT OF NEW YORK

MICHAEL PHILIP KAUFMAN, Plaintiff, v. MICROSOFT CORPORATION, Defendant.	Case No. 16-CV-02880-LTS

DEFENDANT MICROSOFT CORPORATION'S CLAIM CONSTRUCTION SURREPLY BRIEF

TABLE OF CONTENTS

Page

TABLE OF AUTHORITIES

Page(s)

Cases

I. INTRODUCTION

Of the eight terms originally proposed for construction, Kaufman's reply indicates that the "relational interdependencies" limitation is no longer disputed and that plain and ordinary meaning applies. Of the remaining terms, two of them—the "defining" and "integrates" limitations—concern substantive disputes that both parties agree warrant construction. (*See infra* Section III.) The others need no construction—a plain and ordinary meaning is sufficient, but Kaufman redrafts the claims with unsupported, confusing, and unnecessary limitations. (*See infra* Sections IV & V.)

In his reply, Kaufman attached the declaration of Dr. Dennis Shasha. Dr. Shasha's proffer of testimony regarding claim construction issues addressed in Kaufman's opening brief is flawed in many respects. Nevertheless, Dr. Shasha agrees with Microsoft's position that the specification does not provide a description for providing a user interface having all of the required functionality "for each" claimed mode. Specifically, the "delete" mode lacks disclosure in the specification and Dr. Shasha agrees the feature is not contained in any of the source code, meaning there is no embodiment with a delete mode disclosed, and further rendering the claims indefinite because they lack sufficient structure in the specification.

II. "RELATIONAL INTERDEPENDENCIES" TERM IS NO LONGER IN DISPUTE

Kaufman now agrees with Microsoft that plain and ordinary meaning is appropriate for the claim term "while observing and enforcing relational interdependencies among data across said tables," which he had identified for construction. In view of this change in position, no construction is necessary and plain and ordinary meaning applies.

III. DISPUTED CLAIM TERMS

A. "... defining a user interface paradigm comprising a set of modes for interacting with a given database table, said modes comprising create, retrieve, update and delete, and a corresponding display format for each mode" (all claims)

Kaufman	Microsoft
For each of the specified modes of operation (create, retrieve, update and delete), the generated UI includes among its provided display formats at least one display format which supports that operation.	". . . defining a set of user interfaces comprising a separate screen or window for each of the operations of creating, retrieving, updating, and deleting data in a given database table"

The parties agree that the claims require the creation of at least one user interface, though they disagree whether more than one user interface is required. Microsoft's construction requires at least four; under Kaufman's construction just one is sufficient. This dispute seems to boil down to the phrase "a corresponding display format for each mode" and whether it requires a separate display format for each mode as part of "a user interface paradigm comprising a set of modes." Microsoft's proposed construction comports with the plain meaning of the terms and, contrary to Kaufman's assertion (D.I. 41, K. Reply 6), the Federal Circuit mandate to clarify claim scope. *Power-One, Inc. v. Artesyn Techs., Inc.*, 599 F.3d 1343, 1348 (Fed. Cir. 2010) (Construed claim terms must "ensure that the jury fully understands the court's claim construction rulings and what the patentee covered by the claims."); *U.S. Surgical Corp. v. Ethicon, Inc.*, 103 F.3d 1554, 1568 (Fed. Cir. 1997) ("Claim construction is a matter of resolution of disputed meanings and technical scope, to clarify and when necessary to explain what the patentee covered by the claims, for use in the determination of infringement.").[1]

[1] Kaufman notes that Microsoft uses the word "separate," rather than its synonym "distinct," to construe the phrase "corresponding display format for each mode." (K. Reply 6, 9.) "Separate" appears in the specification 12 times. (*See generally* '981 Patent.) Kaufman admits his original argument applies to Microsoft's proposed construction, so it is unclear why Kaufman is raising this issue when he has also revised his constructions, presumably also to narrow and clarify

In contrast, Kaufman ignores the words "a user interface paradigm comprising a set of modes" and "for each," thereby reading out the plain meaning of the claim terms. (K. Reply 6-9.) Indeed, under Kaufman's one-to-many argument, the claim only requires a single display for all modes, but this ignores the words "a set of modes," which is plural, and "a corresponding display format for each," which indicates a distinction with respect to the preceding "set of modes." [2] A single "display" is not a "set of modes," it is "modeless" as the specification expressly distinguishes. ('981 Patent, 11:14-22.)

Kaufman cherry picks the specification to support his argument. (K. Reply 7 (citing to '981 Patent, 12:24-27).) The specification discloses different "modes" (browse, search, add, edit) of interfaces from the four that are claimed (create, retrieve, update, delete).[3] (*See, e.g.,* '981 Patent 3:43-47, Figs. 1-4, 8.) Kaufman's argument that, among other things, the specification "is clear that all modes are supported among the complement of *displays*" (K. Reply 7 (emphasis added)) is confusing. What are the displays (plural) he refers to if there is only one display format, and how could it be "a user interface paradigm comprising *a set of modes*" (where "a set of modes" is plural) if there is only one mode or one display format?

Kaufman's answer is to tacitly concede that two different embodiments are disclosed—"a user interface paradigm with a set of modes" embodiment, which follows Microsoft's

issues before this Court. (*See, e.g.,* K. Reply 5 ("wherein" clause), 18 ("managing . . . said relationships across tables").)

[2] Kaufman's construction—among other issues—eliminates the word "corresponding" from the limitation.

[3] While the '981 patent claims priority to an application family from 2001, Kaufman first claimed a combination of four modes (create, retrieve, update, delete) on October 26, 2007. (*See* Ex. 6, '981 patent history, Remarks Made in Amendment p. 7 (Oct. 26, 2007), MPK0000383.) This claimed combination is not found anywhere in the original application family or the '981 specification.

construction, and an alternative "modeless" embodiment, which is not covered by Microsoft's construction. (K. Reply at 6–9.) Kaufman argues that "[t]here is no statement or even suggestion that the invention does not include a 'modeless' design, and indeed the specification states the opposite." (K. Reply at 8.) This is wrong. The claim language is "a set of modes," which is contrasted with a "modeless" user interface paradigm in the specification:

> A set of "modes" for interacting with a(ny) given database table (which modes, taken together, cover all desired end-user operations which may be effected upon such tables), and a corresponding display format ("screen" or "window" architecture) for each mode.

('981 Patent, 4:65-5:9.)

> Note, finally, that while the preferred embodiment operates according to the particular paradigm described above, . . . it may be desirable in some instances to realize instead a "modeless" UI paradigm, such that all end-user activities (browsing, searching, editing, adding) are supported by a single, unified display context (such as a "spreadsheet" display).")

(*Id.* at 11:14-22.)

Basic rules of English grammar teach that the suffix "-less" means "without" or "lacking." Thus "modeless" means there is no "mode." The claim language thus does not cover the modeless design because it is contrary to having multiple (e.g., at least four) modes. *Intamin Ltd. v. Magnetar Techs., Corp.*, 483 F.3d 1328, 1337 (Fed. Cir. 2007) ("[A] claim need not cover all embodiments."); *Elekta Instrument S.A. v. O.U.R. Sci. Int'l, Inc.*, 214 F.3d 1302, 1308 (Fed. Cir. 2000) ("[T]he unambiguous language of the amended claim controls over any contradictory language in the written description."); *see also Unique Concepts, Inc. v. Brown*, 939 F.2d 1558, 1562–63 (Fed. Cir. 1991) ("subject matter disclosed but not claimed in a patent application is dedicated to the public"). The claim language and basic grammar confirm the patentee's intent to require that each claimed mode has its own display format. The intrinsic record confirms the same, at least for the modes that are disclosed (browse, search, edit and add). ('981 Patent, 4:65–

5:9; 5:27-29 ("MODE-NAVIGATION BAR 710") and FIG. 7 (showing how the screens change for each mode by selecting a mode in the mode-navigation bar 710).) Thus, a plain understanding of the claim language means there is "a separate screen or window for each of the operations of creating, retrieving, updating, and deleting data in a given database table." As Kaufman elected to write them, the claims cannot cover a user interface that is "modeless" and a user interface paradigm with a "set of modes" at the same time.

The parties' dispute is best exemplified in the claimed "delete" mode screen. (*Compare* D.I. 40, MS Br. 10, *with* K. Reply 9.) The '981 patent claims a "delete" mode screen but only discloses a "delete capability." Kaufman's admits that the claimed "delete" mode screen is not supported by the specification, and because of this gap in disclosure argues that the "delete capability" (or operation) is incorporated into the "retrieve" mode screen. (K. Reply 9, n.1.) The claims do not only require a "delete" operation, but a "user interface paradigm comprising a set of modes . . . and a corresponding display format for each mode." If Kaufman wished to only claim create, retrieve, update, and delete ("CRUD") "capabilities" he could have, but he chose not to. Kaufman further conflates Microsoft's "delete" mode screen argument with the disconnect between the specification and the claimed "four standard modes of data operation." (K. Reply 9.) As discussed above, the specification discloses four modes that are different from the four ultimately claimed. None of these four modes disclosed in the specification correspond to the claimed "delete" mode, which Kaufman admits in his brief. (K. Reply at 9; *see* Ex. 8, Shasha Dep. 22:2-19 (Feb. 22, 2017).)

Finally, Kaufman's reliance on expert testimony is misplaced. (K. Reply 9.) Rather, Dr. Shasha's declaration highlights the inconsistency between Kaufman's constructions and the claim language. While Dr. Shasha asserts, with scant analysis, that the code in the specification covers *all* claimed embodiments (D.I. 41, Shasha Decl. ¶ 36 ("the '981 Patent provides full, working source code - disclosing all necessary algorithms - to carry out the processes claimed")), his conclusions are incomplete. First, he does not say which constructions he is applying. Second, he admits that code for a delete operation (which he calls the "Delete control") is not depicted in the patent. (*Id.* ¶ 35.) And third, he is silent on whether a separate screen for the delete mode is disclosed. (*Id.* at ¶ 34 (silent on the delete mode) and ¶ 35 (silent on a separate screen for the delete mode, but instead says it is a button incorporated the "Edit-mode screen.").)

Dr. Shasha admitted in his deposition that the code does not disclose a user interface for the delete mode (Shasha Dep. 19:17–22:19, 24:9–25:8, 37:8-13, 68:18–69:18) and that his conclusions on the sufficiency of the source code depend on Kaufman's proposed constructions (*id.* at 90:4-10, 91:22–92:7; *see also id.* at 52:11–53:2), which do not require a screen for the delete operation. In fact, Dr. Shasha was not even able to perform a delete *operation* in the alleged embodiment offered as Exhibit 2, which contradicts his assertion that code discloses all claimed embodiments—even under Kaufman's constructions:

> Q. Were you able to do a delete operation?
>
> A. No.
>
> Q. Did you try to do a delete operation?
>
> A. Actually, I think -- during that -- I don't think I tried to do a delete operation. I don't think so.
>
> Q. Do you know if a delete operation was possible?
>
> . . .
>
> THE WITNESS: A delete operation with this version of the system I don't think is possible. I don't see any option for it. I don't think I ever tried it because I didn't see any option for it. And in the specification, it says that it's not available, so it's inconsistent.

(Shasha Dep. 22:2-19; *see also id.* at 31:3–35:4, 35:23–37:13.) At bottom, Dr. Shasha's conclusions on the sufficiency of the alleged structure in the specification and source code are not credible because he admits that a user interface for the delete mode is not disclosed.

B. "... integrates into each said mode display processes for representing, navigating, and managing said relationships across tables, for selecting among said modes, and for navigating across said tables and interacting in accordance the selected mode with the data in the tables that are reached by said navigation, while observing and enforcing relational interdependencies among data across said tables" (all claims)

Kaufman	Microsoft
Integrates into each said mode display one or more processes for representing, navigating, and managing said relationships across tables, selecting among said modes for tables reached by said navigation, and interacting in accordance with the selected mode with the data in the tables that are reached by said navigation, while observing and enforcing relational interdependencies among data across said tables. The mode displays, taken together, provide the full complement of such processes.	". . . integrates into each distinct user interface comprising a screen or window processes for (1) representing, (2) navigating, and (3) defining the relationship between one table and another; for selecting among the user interfaces; and for navigating across said tables and interacting according with the selected user interface with the data in the tables that are reached by said navigation, while observing and enforcing relational interdependencies among data across said tables"

Much of Kaufman's reply is repetitive of his arguments regarding Microsoft's construction for the separately raised claim term "managing . . . said relationships across tables." (*Compare* K. Reply 10–12, *with* K. Reply 16–18; *see infra* Section IV.D.) Microsoft's construction for the "integrates" limitation incorporates the plain meaning of the phrase "said relationships across tables," as disclosed in the specification and explained by Kaufman during prosecution of the '981 patent.[4] Kaufman agrees that the three claimed processes (representing, navigating, and managing) are directed to the data model relationships across tables. (K. Reply

[4] During prosecution, Hecht was distinguished as not disclosing a user interface. (Ex. 7, '066 patent history, Response to Office Action p. 7 (Dec. 12, 2006), 7_000086; Ex. 6, '981 patent history, Remarks Made in Amendment p. 7 (Oct. 26, 2007), MPK0000387.) It is unclear how Microsoft's proposed construction for "said relationships across tables" is counter to the prosecution history. (*See* K. Reply 10.)

14.) The ’981 patent says that the “data model” is synonymous with “schema,” and points to the importance of building a front-end application through which “end-users can manipulate” the data model. (’981 Patent, 2:41-48.) And even Kaufman now agrees that a “schema” itself is not the “data,” but rather the schema defines the “relationships” in the database, thus making it “relational.” (K. Reply 2 (defining “schema” to mean “the metadata descriptions of the column complements of the various tables comprising the database and the relationships among the tables”); Shasha Decl. ¶ 21.) Thus, these processes concern doing something with the metadata of various tables—their relationships—as opposed to the data.

Kaufman also fails to respond to questions Microsoft raised about his proposed construction. Kaufman’s construction adds limitations, specifically “one or more,” “taken together,” and “full complement,” without sufficient explanation. In deposition, Dr. Shasha offered that Kaufman’s construction means that for any of the four claimed modes (create, retrieve, update, and delete), only one of the three claimed processes (representing, navigating, managing) needs to be present. This view of the claims is contrary to the express claim language, which refers to plural “display processes” for three enumerated functions that are “integrate[d] into each said mode.” The claim language itself contradicts Dr. Shasha’s extrinsic, unsupported conclusions. *Phillips v. AWH Corp.*, 415 F.3d 1303, 1317 (Fed. Cir. 2005) (“[W]hile extrinsic evidence ‘can shed useful light on the relevant art,’ we have explained that it is ‘less significant than the intrinsic record in determining “the legally operative meaning of claim language.”’” (internal citations omitted)); *Bell Atl. Network Servs., Inc. v. Covad Commc’ns Grp., Inc.*, 262 F.3d 1258, 1269 (Fed. Cir. 2001) (“[E]xtrinsic evidence may be used only to assist in the proper understanding of the disputed limitation; it may not be used to vary, contradict, expand, or limit the claim language from how it is defined, even by implication, in the specification or file history."); *see also Summit 6, LLC v. Samsung Elecs. Co.*, 802 F.3d 1283, 1290 (Fed. Cir. 2015) (Extrinsic evidence may not be used "to contradict claim meaning that is unambiguous in light of the intrinsic evidence." (citing *Phillips*, 415 F.3d at 1324)). Constructions must be based on the claim as written, not as the patentee wishes he had written it. *Chef Am., Inc. v. Lamb-Weston, Inc.*, 358 F.3d 1371, 1374 (Fed. Cir. 2004); *see also Thorner v. Sony Computer Entm't Am. LLC*, 669 F.3d 1362, 1367 (Fed. Cir. 2012) (The claims "define the metes and bounds of the patentee's invention.").

Relatedly, Kaufman argues that Microsoft's construction would read out every embodiment disclosed in the specification. (K. Reply 12, 14.) The claim language "into each" is unambiguous and is therefore not amenable to Kaufman's construction. Kaufman's construction ignores basic grammar, i.e., the plural modes and "display processes," and the distinctive language "into each" and "for each" in the claim itself. This explains why Kaufman tries to redraft the claims. Whether, as Kaufman asserts, all embodiments would be read out of the claim is a consequence of the claim language, and undoubtedly Kaufman's tune will change if he loses his claim construction argument.

Notably, Kaufman walks back his admission that the "retrieve" mode lacks the "managing" process. (K. Reply 12–13; *see* D.I. 39, K. Br. 15 ("the 'Retrieve' mode displays of the specification, Figs. 1 and 2, provide for representing and navigating across tables, and selecting among modes, but not for 'managing said relationships across tables.").) For support, Kaufman now relies on portions of the specification that do not specifically address the "managing" process in the "retrieve" mode, but generally relates other functions. (*Compare* K. Reply 13 (citing to '981 Patent, 5:54-56, 6:57–7:40), *with* '981 Patent, 5:10-11 (prefacing the subsequent key screen elements "for navigation control/support are shared across all of these displays"), *and* 6:46-56 (explaining the rules and methods that follow are "for extending the representation of any single table (according to its relationships to other tables) (FIGS. 7 and 8), and for managing (and navigating across) these relationships").) At bottom, Kaufman cannot read out a claim term because it is "unnecessary"; Kaufman has to live with the claims as drafted. This is the bargained-for exchange of the patent grant, and to redraft the claims deprives the public of notice in derogation of that bargained-for exchange.

IV. "WHEREIN SAID RELATIONAL DATABASE MAY BE OF ANY ARBITRARY SIZE AND COMPLEXITY" CLAUSE

Kaufman (Revised[5])	Microsoft
The database, consisting of both a schema~~, rather than the~~ and data~~ within the database~~, may be of any arbitrary size or complexity	The claim term lacks support in the specification, otherwise, plain and ordinary meaning.

After changing his construction, Kaufman's dispute over this limitation seems to be some confusion about what a "relational database" is. Kaufman's expert Dr. Shasha agrees with Microsoft that "relational database" is a term well known in the art and understands its plain and ordinary meaning to "refer to both the database schema as well as the data." (Decl. of Dennis Shasha ¶ 21, D.I. 41; Shasha Dep. 15:21-23.) Thus, no construction is needed beyond the plain and ordinary meaning—the words of the claim are sufficient.

[5] Deletions to Kaufman's original construction appear in strikethrough and additions appear in underline.

V. KAUFMAN IMPROPERLY IMPORTS LIMITATIONS INTO DISPUTED CLAIM TERMS

A. "automatically generating" (all claims)

Kaufman	Microsoft
The "generating" - comprising steps (a), (b), and (c) - is carried out upon being triggered by a user, without requiring further intervention by the user in order to complete the generation of the UI.	Plain and ordinary meaning.

Instead of explaining what his proposed "triggering event" is, Kaufman mischaracterizes Microsoft's "plain and ordinary meaning" position, generally, and Microsoft's "user intervention" argument, specifically. (K. Reply 5.) There is nothing unique to the claim terms "automatically generating" that warrants deviating from their plain and ordinary meaning. The claim terms are common words that are easily understood. Kaufman, however, proposes replacing the ordinary word "automatically" with over two dozen words in a construction that (1) is unclear, (2) adds extraneous limitations, and (3) lacks support or explanation. For example, the phrase "*without requiring* further intervention" in Kaufman's proposed construction is not clear—why is user intervention even mentioned and what does it mean here? Does Kaufman really mean that user intervention is prohibited? Kaufman refuses to acknowledge this issue in his proposed construction. (*See id.* (arguing that Microsoft, by acknowledging the plain meaning permits some user intervention, is "seek[ing] to define away one of the clear advances of Kaufman's invention"—though he does not say what that advancement is or explain why it was not specified in the claims as written).)

Kaufman maintains that his construction precludes "custom programming," but fails to explain the relationship, if any, between "custom programming," "triggered by a user," and "requiring further intervention by the user." (K. Reply 5; *see* K. Br. 10–11.) And Kaufman proffers no explanation as to his proposed "triggering event." (*See* K. Reply 5.) Further, the specification provides no definition or discussion of "custom programming." (*See generally* '981 Patent.) Kaufman argues that no "custom programming" refers to the creation of "specific code or instructions for each table or relationship." (K. Reply 5.) This "custom programming" argument is unsupported by Dr. Shasha's declaration. Dr. Shasha argues that "automatically generating" relates to "the manipulation and retrieval of the data within the database, rather than the problem of designing the schema itself." (Shasha Decl. ¶ 26.) Dr. Shasha's statement does not reflect language in Kaufman's construction, nor link that construction to claim language or the specification, further undermining the relevance of his conclusions. Dr. Shasha also makes no correlation between "automatically generating" and "custom programming." And Dr. Shasha fails to explain why "the manipulation and retrieval of *data* within a database" is treated differently from the *schema* and how it is relevant to Kaufman's construction. (Shasha Decl. ¶ 26 (emphasis added).) Under Kaufman's construction, limitations (a)-(c) are all "generat[ed]" "without any custom programming." But, limitations (a)-(c) are not limited to "the manipulation and retrieval of data within a database," nor are they all limited to "specific code or instructions for each table or relationship." Thus, Kaufman's construction is unsupported by both the specification and his expert.

B. "representing . . . said relationships across tables" (all claims)

Kaufman	Microsoft
Visually representing at least the existence of a relationship from the data record(s) in the table being viewed to the corresponding data in another table (if any) that is related to the viewed-table data record(s), according to the data model for the database.	Plain and ordinary meaning, consistent with the above construction, i.e., "representing . . . the relationship between one table and another"

Kaufman admits there is a disconnect between the claim language and the alleged invention. (K. Reply 14–15.) For that reason, Kaufman proffers a different construction of "said relationships across tables" for each of the "representing," "navigating," and "managing" processes. But this admitted disconnect does not give Kaufman free reign to construe one term—appearing in one place in the claims—to mean three different things. *Frank's Casing Crew & Rental Tools, Inc. v. Weatherford Int'l, Inc.*, 389 F.3d 1370, 1377 (Fed. Cir. 2004). To be clear, Microsoft's issue is not—as Kaufman describes—that "Kaufman used different words to describe each of representing, navigating, and managing." (*See* K. Reply 14.) Rather, Kaufman is proposing three very different meanings for *one term*—"said relationships across tables." In essence, Kaufman is using claim construction to redraft the claims and create three very different functions than what is plainly claimed.

For "representing . . . relationships across tables," Kaufman argues—without any support from the specification—that "the relationships of the data model" are "manifested in the data that is stored in the database" and "represented in the display of one database table by a visual representation of related data that resides in another table." (K. Reply 15.) What exactly this means is anyone's guess; it seems to suggest a non-relational database can be a relational database so long as the *data* in one table can, in theory, somehow relate to the *data* in another table. Kaufman's construction improperly reframes the limitation to include the data in a table when the term "data" does not even appear in the limitation. In contrast, Kaufman's expert seems to agree with Microsoft that "representing . . . relationships across tables" refers to "representing . . . the relationship between one table and another." (Shasha Decl. ¶ 16 ("The 'City' link . . . represents that there is a relation between the State or Province and City tables.").)

Further, Kaufman fails to address the subjective language—"at least the existence of" and "if any"—he proposes. (*See* K. Reply 14–15; MS Br. 15.) The claim limitations are not conditional, and subjective claim language and the absence of objective boundaries introduce

Section 112 definiteness issues. *Interval Licensing LLC v. AOL, Inc.*, 766 F.3d 1364, 1371–73 (Fed. Cir. 2014). Construing claim terms should clarify issues, not create new ones.

Last, Kaufman agrees that his proposed "visually" limitation lacks direct support from the specification so he is forced to rely on the specification's discussion of "screens" and "displays" to support his construction. (K. Reply 14.) *See Hill-Rom Servs., Inc. v. Stryker Corp.*, 755 F.3d 1367, 1371 (Fed. Cir. 2014) (departure from plain and ordinary meaning based on lexicography in the specification requires the patentee "clearly set forth a definition of the disputed claim term other than its plain and ordinary meaning" and "clearly express an intent to redefine the term." (internal citation omitted)). Kaufman's argument is inconsistent with his other positions, at least so far as he disagrees with Microsoft's "integrates" construction, which construes "mode display" to mean (in relevant part) "screens or windows," yet here he seems to acknowledge the same idea is required.

C. "navigating . . . said relationships across tables" (all claims)

Kaufman	Microsoft
For the visually represented cross-table relationships, navigating from the data in the viewed table to a display of the data in a related table that is specifically related to the viewed-table data record.	Plain and ordinary meaning, consistent with the above construction, i.e., "navigating . . . the relationship between one table and another"

Kaufman's construction for a phrase—"said relationships across tables"—that is paired with three functions in the claims—"navigating," "representing," and "managing"—differs from function to function. (*Supra* Section V.B.) Kaufman does not address how the additional limitations in his construction for the phrase "across tables" comports with the canons of claim construction. *See, e.g., Hill-Rom Servs., Inc. v. Stryker Corp.*, 755 F.3d 1367, 1371 (Fed. Cir. 2014) (applying plain and ordinary meaning based on the specification except for instances of lexicography and disavowal); *Frank's Casing Crew*, 389 F.3d at 1377 (terms are construed "in a manner that renders the patent internally consistent."); *Home Diagnostics, Inc. v. LifeScan, Inc.*, 381 F.3d 1352, 1359 (Fed. Cir. 2004) (additional limitations should not be imported into a claim); *Chef Am.*, 358 F.3d at 1374 (claims are construed as drafted). In support of this construction, Kaufman proffers the same narrow example from his opening brief to illustrate this claim term, but fails to explain why new limitations, such as "the visually represented cross-table relationships" and "data in the viewed table," are necessary to either accord or depart from this claim term's plain and ordinary meaning. (*See* K. Reply 15.)

Kaufman's reliance on the specification to support his proposed construction is conclusory. He does not say what a "cross-table relationship" or "viewed-table data record" are or why these specially coined terms in his construction are better than the plain language of the claim. Kaufman points to two of the "key screen elements for navigation control/support . . . shared across all of [the claimed] displays," but fails to explain why "Hot Link 806" and "Hot Link 808" are more significant than "Scroll Navigation 706." (*Compare* '981 Patent, 5:10-62, *with* K. Reply 15 (citing to '981 Patent, 5:50-53).) Similarly, Kaufman calls out one of the "rules and methods for traversing/navigating the context stack," but fails to connect this single rule and method to the claim language or his proposed construction. (*Compare* '981 Patent, 9:8–11:6, *with* K. Reply 15 (citing to '981 Patent, 9:23-30).) Again, Kaufman fails to explain why his alleged illustration of "navigating" as a "drill-down" departs from the plain and ordinary meaning of the term, let alone the broad understanding of the word as disclosed in the specification.

D. "managing . . . said relationships across tables" (all claims)

Kaufman (Revised)	Microsoft
Managing the relationships of records in a given table with corresponding records in a related table~~, for example, by way of a dropdown that limits~~ by limiting selection of an added or edited value for a record in the given table to the permissible values as exist within the records of the related table.	Plain and ordinary meaning, consistent with the above construction, i.e., "defining the relationship between one table and another"

Kaufman vaguely asserts that Microsoft's proposed construction would "make meaningless the express claim term" and "define away Kaufman's contribution to the art" (K. Reply 10, 16), but Kaufman later agrees that the portions of the specification Microsoft relies on are about "managing." (K. Reply 16–18.) Specifically, Kaufman argues the "managing" limitation is disclosed the specification at col. 5:54-56, 6:46-7:40.[6] (K. Reply 13, 17, 18.) These portions of the specification discuss the "key screen element" called "cross-reference field" and a "set of rules and methods for . . . represent[ing] . . . managing (and navigating) across these relationships" including "foreign-key fields," respectively. Microsoft relies on the specification at col. 6:48-56 as well as the discussion before that to support its construction. These portions of the specification, as well as col. 24:9-19, discuss cross-referencing, one of the actions disclosed in the specification that, when viewed together, amount to "defining" the relationship between the tables. Thus, Kaufman's issue seems not to be whether "managing" in this context means "defining," but Microsoft's refusal to redraft the claim language (i.e., "said relationships across tables") in order to redirect the limitation to a completely different meaning focused *what* is *in* a

[6] Under Kaufman's new construction, it is unclear whether "'[m]anaging' means managing" or "limiting selection." (K. Reply 16, 18.)

given table.[7] In other words, Kaufman argues that managing said *relationships* between tables does not mean doing something to affect or change the *relationships* across tables*,* but rather entering or changing *data* in a single table. The problems with Kaufman's arguments are: (1) the claims do not say "managing data" they say "managing said relationships"; (2) the "create," "update" and "delete" modes already manage data (Kaufman's view), since these three modes cover all operations that would change the data, thus making the "managing said relationships" limitation superfluous; and (3) if Kaufman's construction was correct, then it fails to account for the "retrieve" mode, since Kaufman admits that the retrieve mode does not manage relationships between tables even under his construction because you would not be changing the data in the retrieve mode. (K. Reply at 13 ("On the other hand, a browse or search screen has no need to perform such managing, because it is retrieving data, not changing it.").)

Furthermore, Kaufman's expert, Dr. Shasha, admitted in deposition that Kaufman's construction is already embodied in the last clause in the claims ("while observing and enforcing relational interdependencies among data across said tables"), making Kaufman's attempt to redefine the limitation an attempt to excise it from the claims. (Shasha Dep. 54:25–56:20.) *Power Mosfet Techs., L.L.C. v. Siemens AG*, 378 F.3d 1396, 1410 (Fed. Cir. 2004) ("[I]nterpretations that render some portion of the claim language superfluous are disfavored."); *see Aristocrat Techs. Australia Pty Ltd. v. Int'l Game Tech.*, 709 F.3d 1348, 1356–57 (Fed. Cir. 2013) (declining to adopt the appellants' proposed construction because it would render another limitation "superfluous"); *Merck & Co. v. Teva Pharm. USA, Inc.*, 395 F.3d 1364, 1372 (Fed. Cir. 2005) ("A claim construction that gives meaning to all the terms of the claim is preferred

[7] Kaufman alleges—without support—that "defining" is an attempt to "redirect the patent toward the prior art." (K. Reply 10.) It is unclear what prior art Kaufman is referring to or how a proposed construction based on the disclosures in the specification merit this characterization.

over one that does not do so.").

Kaufman's new construction redirects the claim away from his express language "managing said relationships across tables," to instead make the claim about how one might enter data into a table. (K. Reply 17–18.) Entering data into a single table is not "managing [the data model] relationships across tables"—there is no change in the schema, and thus no change in the relationship of one table to another would result. (*See, e.g.,* '981 Patent, 2:41-48, 3:26-32, 11:40-45, 11:50-56.) The three processes (representing, navigating, and managing) are directed to the *relationships* between tables in the relational database. The three processes do not refer to "data." Kaufman fails to reconcile the claim language and intrinsic evidence with his proposed construction and it should be rejected.

VI. INDEFINITENESS

Williamson v. Citrix Online, LLC is about fairness and accountability—acknowledging that a claim may not look like a "bare" function but may be just that and holding patentees accountable for the words they use to describe their invention.[8] 792 F.3d 1339, 1349 (Fed. Cir. 2015) (en banc) (noting that the now-overruled heightened burden "resulted in a proliferation of functional claiming untethered to § 112, para. 6 and free of the strictures set forth in the statute"). *Williamson* puts patentees on notice that not every software-related patent will be subject to interpretation under Section 112, paragraph 6 provided that the claims are properly drafted, that is, computer-implemented functional claims must have corresponding disclosures setting forth algorithms to perform the claimed functions. *Id.* at 1352. The '981 Patent includes computer-implemented functional claims.

[8] Subjecting a functional claim to Section 112, paragraph 6 is not a death sentence, as Kaufman suggests. (K. Reply 20, 23.) A claim subject to Section 112, paragraph 6 is only as narrow as the disclosure it corresponds to. It is the responsibility of the patentee to make sure that the broad functional claims he writes have sufficient support.

The ’981 Patent is proof that a “highly technical” specification that “consists largely of computer code” (and is purportedly an “exhaustive and complete recitation of source code”) can still fail to sufficiently disclose the structure for the claimed functions. Kaufman’s generic assertion that all of the structure is somewhere within 150 pages of source code is belied by Dr. Shasha’s admissions that: (1) he was unable to perform a delete operation using the alleged embodiment of the patent; (2) that the delete mode interface is not disclosed; and (3) that all three functionalities (i.e., representing, navigating, and managing) are not disclosed for each mode. Absent sufficient structure, the ’981 Patent claims are indefinite.

A. Claims 1, 4, and 5 recite functionality and the desired result

Claims 1, 4, and 5 recite “*what* to do, but not *how* to do it,” and are therefore subject to Section 112, paragraph 6.[9] (*See* K. Reply 22.) As to claims 4 and 5 (the two apparatus claims), Kaufman offers no response to treating “machine-readable routines” and “routine for” as nonce words for performing functions. (*See* K. Reply 22, 23.) The claim language in limitations (a), (b), and (c) that follow these nonce words recite functionality and the desired result of that functionality.

Instead of addressing all three independent claims (two of which are apparatus claims), Kaufman attacks the method claim (K. Reply 23), but even that attack misses the mark because limitations (a), (b), and (c) of claim 1 are directed to the same functionality and desired results as claims 4 and 5. Thus, the example Kaufman uses to purportedly show how the independent claims recite “algorithmic specifics” that take the claims outside the realm of § 112 ¶ 6 functional claiming proves the opposite. (K. Reply 23–24.) What Kaufman identifies are not

[9] *Williamson* does not set forth an “entirely functional” requirement. (*Cf.* K. Reply 22.) The presumption is overcome upon a showing that “the claim term fails to ‘recite sufficiently definite structure’ or else recites ‘function without reciting sufficient structure for performing that function.’” *Williamson*, 792 F.3d at 1349.

"operations" but a few functions and the result of those functions (but not how the results are achieved):

Kaufman's Limitation (c) "Operations"	Functionality Claimed
construct[ing] a corresponding client application [i.e., corresponding to the "data model"],	• "constructing" is a function • the result is a "client application"
[the] client application provid[ing] a connection to [the] database	• "providing" is a function • the result is connecting to the database
provid[ing] displays of the table contents of [the] database for each of [the CRUD] modes	• "providing" is a function • the result is the display of data
Integrating into each mode display:	• "integrating" is a function • the results are in the other so-called "operations"
processes for representing, navigating, and managing said relationships across tables,	• "processes" is another word for "algorithm" • "processes *for*" indicates that what follows is the result of the algorithm, not rules making up the algorithm
processes for selecting among said modes	• "processes" is another word for "algorithm"; • "processes *for*" indicates that what follows is the result of the algorithm, not rules making up the algorithm
processes for navigating across said tables and interacting in accordance the selected mode with the data in the tables that are reached by said navigation	• "processes" is another word for "algorithm"; • "processes *for*" indicates that what follows is the result of the algorithm, not rules making up the algorithm
all while observing and enforcing relational interdependencies among data across said tables	• "observing" and "enforcing" are functions • how they are observed and enforced is not specified

Kaufman fails to identify a set of rules or processes—within the claims—by which these functions are achieved. *See Media Rights Techs., Inc. v. Capital One Fin. Corp.*, 800 F.3d 1366, 1374 (Fed. Cir. 2015) (requiring that all of the functions associated with the claim at issue be supported by the disclosed algorithm(s)). The foundation of a method claim is its disclosure of "how" to do something, but as shown above with respect to limitation (c), claim 1 does not tell the public "how" to make a "client application," it only tells the public "what" is included in a "client application." Kaufman only addresses limitation (c), but the same problem exists for limitations (a) and (b)—there is no "how," just a lot of "what."

B. The source code does not provide sufficient structure

The mere existence of source code cannot satisfy the requirements of § 112 ¶ 6. Kaufman points generally at 150 pages of code included in the specification (K. Reply 22, 24), but the question under *Williamson* is what structure corresponds to the functionality at issue. As shown above, with respect to limitation (c), the claims recite numerous functions, each of which must have corresponding structure. *See Media Rights Techs.*, 800 F.3d at 1347 (finding two of the four functions performed by the means-plus-function term lacked sufficient corresponding structure in the specification). Instead, Dr. Shasha only addresses one function for each of the limitations in question.

Kaufman also relies on a supposed "working example" generated by the source code of unknown provenance, which purportedly embodies the claims.[10] For this Kaufman relies on Dr. Shasha's declaration, who does not attest to the provenance of the code or the embodiment itself. Dr. Shasha did not setup the computer or even take the screen captures in his report, which appear to have occurred the evening before his declaration was submitted. (*See* Shasha Dep. 19:22–21:12, 24:9-14, 37:22–38:24, 39:17–44:4.) And as noted above, Dr. Shasha's conclusions

[10] Exhibit 2 to Dr. Shasha's declaration was not created using the source code produced under the local patent rules, and no declaration verifying its authenticity, date and circumstances of its assembly was provided. Plaintiff's counsel first claimed the code was work product then produced it after Dr. Shasha's deposition. Kaufman's reliance on this working example is untimely and improper under the relevant patent rules. *See* D.I. 32 ¶ 4; N.D. Cal. L.P.R. 3-1(g) and 3-2(e).

are completely dependent on Kaufman’s claim constructions—if they are incorrect, then the code does not embody the claims, but even if all of Kaufman’s constructions are correct, Dr. Shasha admitted that the delete mode is not supported by the code. (*Id.* at 22:2-19, 68:18–69:18.)

1. Shasha fails to identify sufficient structure for all functions claimed in limitation (a)

Limitation (a) includes at least two functions: (1) “providing an output stream for user display and input devices” and (2) “defining a user interface paradigm” comprising a certain set of modes: create, retrieve, update and delete. Dr. Shasha explains that the source code he reviewed “contains many statements for emitting such an output stream,” but fails to identify such “statements” with any specificity. (Shasha Decl. ¶¶ 29–30.) Further, Dr. Shasha fails to completely identify source code corresponding to the second function. Dr. Shasha merely notes that the source code produced in the specification indicates that “BROWSING” may be “rendered on a user’s web browser.” (Shasha Decl. ¶ 30.) “BROWSING” is not a claimed mode of the user interface paradigm. Dr. Shasha did not identify source code showing each mode—specifically the delete mode—may be rendered on a user’s computer in accordance with the claims. (Shasha Dep. 21:19–22:19, 37:8-13, 68:18–69:18.) This is at least consistent with Kaufman’s admission that the specification does not support a delete mode display. (K. Reply 6–7, 9.)

To try get around this problem, Dr. Shasha claimed in deposition that a person of ordinary skill in the art at the time of the alleged invention would be capable of writing an algorithm to perform the claimed function (Shasha Dep. 70:16–71:17), but that is contrary to what the law requires, which is disclosure of the *specific* algorithms.[11] *ePlus, Inc. v. Lawson Software, Inc.*, 700 F.3d 509, 519 (Fed. Cir. 2012) ("The indefiniteness inquiry is concerned with whether the bounds of the invention are sufficiently demarcated, not with whether one of ordinary skill in the art may find a way to practice the invention."); *see Function Media, L.L.C. v. Google, Inc.*, 708 F.3d 1310, 1318 (Fed. Cir. 2013) ("When dealing with a 'special purpose computer-implemented means-plus-function limitation,' we require the specification to disclose the algorithm for performing the function."); *Blackboard, Inc. v. Desire2Learn, Inc.*, 574 F.3d 1371, 1385 (Fed. Cir. 2009) ("A patentee cannot avoid providing specificity as to structure simply because someone of ordinary skill in the art would be able to devise a means to perform the claimed function." (internal citations omitted)); *Aristocrat Techs. Australia Pty Ltd. v. Int'l Game Tech.*, 521 F.3d 1328, 1337–38 (Fed. Cir. 2008) (finding terms lacked sufficient structure where a person of ordinary skill in the art would not recognize the patent as disclosing the required algorithm). Thus, limitation (a) is indefinite for lacking sufficient corresponding structure in the specification, including the 150 pages of source code—all of the claimed functionality must be enabled through sufficient structure and algorithms, which even Kaufman admits are not present, particularly as it pertains to the delete mode.

2. Shasha fails to identify sufficient structure for all functions claimed in limitation (c)

Kaufman identifies at least eight functions included in limitation (c). (*See supra* Section V.A; K. Reply 23–24.) Kaufman further relies on Dr. Shasha to "explain[], with direct support

[11] Dr. Shasha defines the level of ordinary skill in the art one way (Shasha Decl. ¶ 13), but admitted that this level of skill would be insufficient to address the indefiniteness inquiry because more knowledge would be required for a person to write the code Dr. Shasha says is needed for the aspirational delete operations. (Shasha Dep. 13:7–14:24, 71:7-17, 87:13–88:14.) This, again, undermines Dr. Shasha's conclusions.

from the specification's source code, how *each claimed operation* is detailed in algorithms set forth in the source code." (K. Reply 25.) Instead, Dr. Shasha lumps all eight functions together. (Shasha Decl. ¶¶ 34, 36.) Dr. Shasha further lumps the entire source code together, arguing that it "as a whole operates" to satisfy these eight functionalities. (*Id.*) This is inconsistent with his admission that "delete" mode is not shown in any figure and is not disclosed as screen display. (*Id.* at ¶ 35 (pointing to a disclosure in the specification referring to a "delete capability" "within the Edit-mode display").) In fact, Dr. Shasha admitted that there is no delete mode disclosed in Exhibit 2, and he was not able to test it. (Shasha Dep. 22:2-19; 68:18–69:18.) Equally important though, Dr. Shasha's analysis is dependent upon his conclusion that "at least one" but not all three claimed functions (i.e., representing, navigating, and managing) are required for each mode. (*Id.* at 68:9-14, 72:22–73:6.) Dr. Shasha admits that not all three functions are shown in the display for each mode. (*Id.* at 73:7-12.) Accordingly, there is insufficient structure in the specification to support the claimed functions of limitation (c).

VII. CONCLUSION

For these additional reasons, Microsoft respectfully requests that this Court adopt their proposed claim constructions and find claims 1, 4, and 5 invalid as indefinite.

Dated: March 1, 2017 Respectfully submitted,

FISH & RICHARDSON P.C.

By: */s/ Leah A. Edelman*
Leah A. Edelman (LE7384)
edelman@fr.com
Ahmed J. Davis (*Pro Hac Vice*)
adavis@fr.com
1425 K Street, N.W., 11th Floor
Washington, D.C. 20005

Telephone: (202) 783-5070

Karolina Jesien (KJ7292)
jesien@fr.com
601 Lexington Avenue, 52nd Floor
New York, New York 10022
Telephone: 212-765-5070

Jason W. Wolff (*Pro Hac Vice*)
wolff@fr.com
12390 El Camino Real
San Diego, CA 92130
Telephone: (858) 678-5070

ATTORNEYS FOR DEFENDANT MICROSOFT CORPORATION

CERTIFICATE OF SERVICE

I hereby certify that on March 1, 2017, I electronically filed the foregoing document with the Clerk of the Court using the CM/ECF system which will send notification of such filing to all attorneys of record.

_ *s/ Leah A. Edelman*______
Leah A. Edelman